

AMENDMENTS TO THE CLAIMS:

Please amend claim 13 as follows.

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (*Original*) A method of processing data with execution of data processing operations under control of either a first operating system or a second operating system, said method comprising the steps of:

receiving an interrupt operable to suspend execution of data processing operations;

in response to said interrupt, starting a stub interrupt handling routine executing under control of said first operating system;

as commanded by said stub interrupt handling routine, suspending execution of said stub interrupt handling routine and starting a main interrupt handling routine executing under control of said second operating system;

executing said main interrupt handling routine under control of said second operating system to handle said interrupt;

as commanded by said main interrupt handling routine, resuming execution of said stub interrupt handling routine under control of said first operating system; and

as commanded by said stub interrupt handling routine, resuming said data processing operations, wherein

if said main interrupt handling routine is interrupted by a further interrupt which when handled leaves processing under control of said first operating system, then said

first operating system detects that said stub interrupt handling routine has been interrupted and resumes said stub interrupt handling so as to trigger resumption of said main interrupt handling routine.

2. *(Original)* A method as claimed in claim 1, wherein when said interrupt occurs while data processing under said first operating system is suspended following data processing operations under control of said first operating system executing a call instruction calling data processing operations under control of said second operating system, said stub interrupt handling routine appears to said first operating system to be handling an interrupt which occurred during execution of said call instruction.

3. *(Original)* A method as claimed in claim 2, wherein said resumption of data processing operations as commanded by said stub interrupt handling routine is performed by re-executing said call instruction.

4. *(Original)* A method as claimed in claim 2, wherein said call instruction is a software interrupt instruction.

5. *(Original)* A method as claimed in claim 1, wherein said second operating system executes in a secure domain and said first operating system executes in a non-secure domain, wherein a data processing operations executing in said secure domain

have access to secure data which is not accessible to a data processing operating executing in said non-secure domain.

6. *(Original)* A method as claimed in claim 1, wherein switches between processing under control of said first operating system and processing under control of said second operating system take place via a monitor mode of operation executing a monitor mode program.

7. *(Original)* Apparatus for processing data with execution of data processing operations under control of either a first operating system or a second operating system, said apparatus comprising processing logic operable to perform the steps of:

receiving an interrupt operable to suspend execution of data processing operations;
in response to said interrupt, starting a stub interrupt handling routine executing under control of said first operating system;

as commanded by said stub interrupt handling routine, suspending execution of said stub interrupt handling routine and starting a main interrupt handling routine executing under control of said second operating system;

executing said main interrupt handling routine under control of said second operating system to handle said interrupt;

as commanded by said main interrupt handling routine, resuming execution of said stub interrupt handling routine under control of said first operating system; and

as commanded by said stub interrupt handling routine, resuming said data processing operations, wherein

if said main interrupt handling routine is interrupted by a further interrupt which when handled leaves processing under control of said first operating system, then said first operating system detects that said stub interrupt handling routine has been interrupted and resumes said stub interrupt handling so as to trigger resumption of said main interrupt handling routine.

8. *(Original)* Apparatus as claimed in claim 7, wherein when said interrupt occurs while data processing under said first operating system is suspended following data processing operations under control of said first operating system executing a call instruction calling data processing operations under control of said second operating system, said stub interrupt handling routine appears to said first operating system to be handling an interrupt which occurred during execution of said call instruction.

9. *(Original)* Apparatus as claimed in claim 8, wherein said resumption of data processing operations as commanded by said stub interrupt handling routine is performed by re-executing said call instruction.

10. *(Original)* Apparatus as claimed in claim 8, wherein said call instruction is a software interrupt instruction.

11. (*Original*) Apparatus as claimed in claim 7, wherein said second operating system executes in a secure domain and said first operating system executes in a non-secure domain, wherein a data processing operations executing in said secure domain have access to secure data which is not accessible to a data processing operating executing in said non-secure domain.

12. (*Original*) Apparatus as claimed in claim 7, wherein switches between processing under control of said first operating system and processing under control of said second operating system take place via a monitor mode of operation executing a monitor mode program.

13. (*Currently Amended*) A computer program product having a computer program operable to control a data processing apparatus in accordance with a method as claimed in ~~any one of claims 1 to 6~~claim 1.